

[EIC0110] Conceção e Análise de aLgoritmos
2010-2011

Trabalho de Grupo 2:

Tema 4

Auto-Complete

Turma 3 – Grupo 13

Maio de 2011

Trabalho realizado por:

Maria Antonieta Dias Ponce de Leão e Oliveira – 070509157 – ei07157@fe.up.pt

Índice

Índice	2
1 – Introdução	3
1.1 – Resumo.....	3
1.2 – Limites da aplicação	3
1.3 – Resultados esperados.....	3
2 – Detalhes da implementação.....	4
2.1 – Situações de contorno.....	4
2.2 – Algoritmos implementados	5
2.3 – Diagrama de classes	6
2.4 – Casos de utilização	7
2.5 – Dificuldades encontradas	8
3 – Manuais	9
3.1 – Manual de instalação num IDE.....	9
3.2 – Manual de instalação	9
3.3 – Manual de utilização	9
4 – Conclusão	11
4.1 – Desenvolvimentos futuros	11
Bibliografia	12

1 – Introdução

1.1 – Resumo

Este trabalho consiste na criação de um editor de texto simples, tipo VI, com auto-complete. Auto-complete é um recurso útil na produção de texto e é encontrado em diversas aplicações como teclados virtuais de telemóveis (T9/XT9) ou ambientes IDE para codificação de programas, recorrendo a teclas de atalho para a selecção d'entre as opções sugeridas pela aplicação. A selecção de palavras será feita com a ajuda de um dicionário, Português de Portugal (por defeito). O editor deverá incluir várias funcionalidades adequadas que permitam a correcta interacção com o programa e uma visualização compreensível dos resultados.

1.2 – Limites da aplicação

O principal limite da aplicação é o facto de ter de ser efectuada em linha de comandos. Poderia implementar-se a aplicação, de forma muito mais interessante, se esta restrição não existisse.

Existem no entanto outras limitações devidas ao limite de tempo tais como:

- Apenas um dicionário estará disponível para cada edição, pode alterar-se o dicionário antes de iniciar a edição de texto, mas para o alterar a meio da mesma é necessário primeiro gravar o ficheiro, alterar o dicionário e abrir o ficheiro previamente guardado;
- Ao sair da edição de texto, se não for gravado um ficheiro, o texto será perdido;
- Não existe a possibilidade de usar cursores para movimentar o cursor no texto;
- Os ficheiros de texto a serem utilizados para leitura deveram ser guardados na pasta 'CAL1011_2MIEIC3_13_TG2_4\Editor\documents', esta será também a pasta na qual os ficheiros serão gravados;
- Não são aceites ficheiros de dicionários em formatos que não ANSI.

1.3 – Resultados esperados

Prevê-se que seja implementado um editor de texto com a funcionalidade de auto-complete, através da pesquisa de palavras num dicionário. Esta pesquisa de palavras deverá ser implementada de forma eficiente tendo em conta a complexidade espacial e temporal do algoritmo implementado.

2 – Detalhes da implementação

2.1 – Situações de contorno

Uma das situações de contorno foi alterar o formato dos dicionários, que não se encontravam no formato ANSI.

Foi alterado o conteúdo do dicionário, removendo as expressões que se encontravam no mesmo, de forma apenas a reduzir o tamanho deste.

Apenas se criaram 9 opções de escolha de cores tanto de texto como de fundo para facilitar os atalhos para as mesmas.

2.2 – Algoritmos implementados

O algoritmo implementado foi baseado numa *N-ary tree*, e consiste no seguinte:

- O dicionário é carregado para uma árvore, em que os nós têm um nome, um nível e uma palavra parcial;
- Uma palavra é introduzida na árvore, letra a letra, em que cada letra irá corresponder a um nó com esse nome, criando-o se não existir;
- A palavra parcial de cada nó será o conjunto de todos os caracteres da palavra até esse nó. Exemplo:

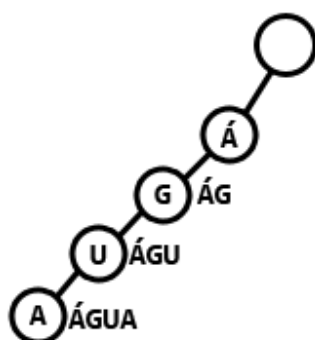


Fig. 1 - Exemplo do processamento na árvore



Fig. 2 - Exemplo de utilização do auto-complete

- Pesquisa-se na árvore a sequência de letras da palavra introduzida até chegar ao fim dessa palavra.
- Retornam-se um número de palavras, previamente definido, correspondentes à sequência introduzida que incluem as palavras desse nó e dos nós filhos.

Este algoritmo tem uma complexidade temporal de $O(N)$, as estruturas de apoio ao algoritmo têm uma complexidade espacial de $O(a.N)$, sendo a o número médio de caracteres por palavra.

2.3 – Diagrama de classes

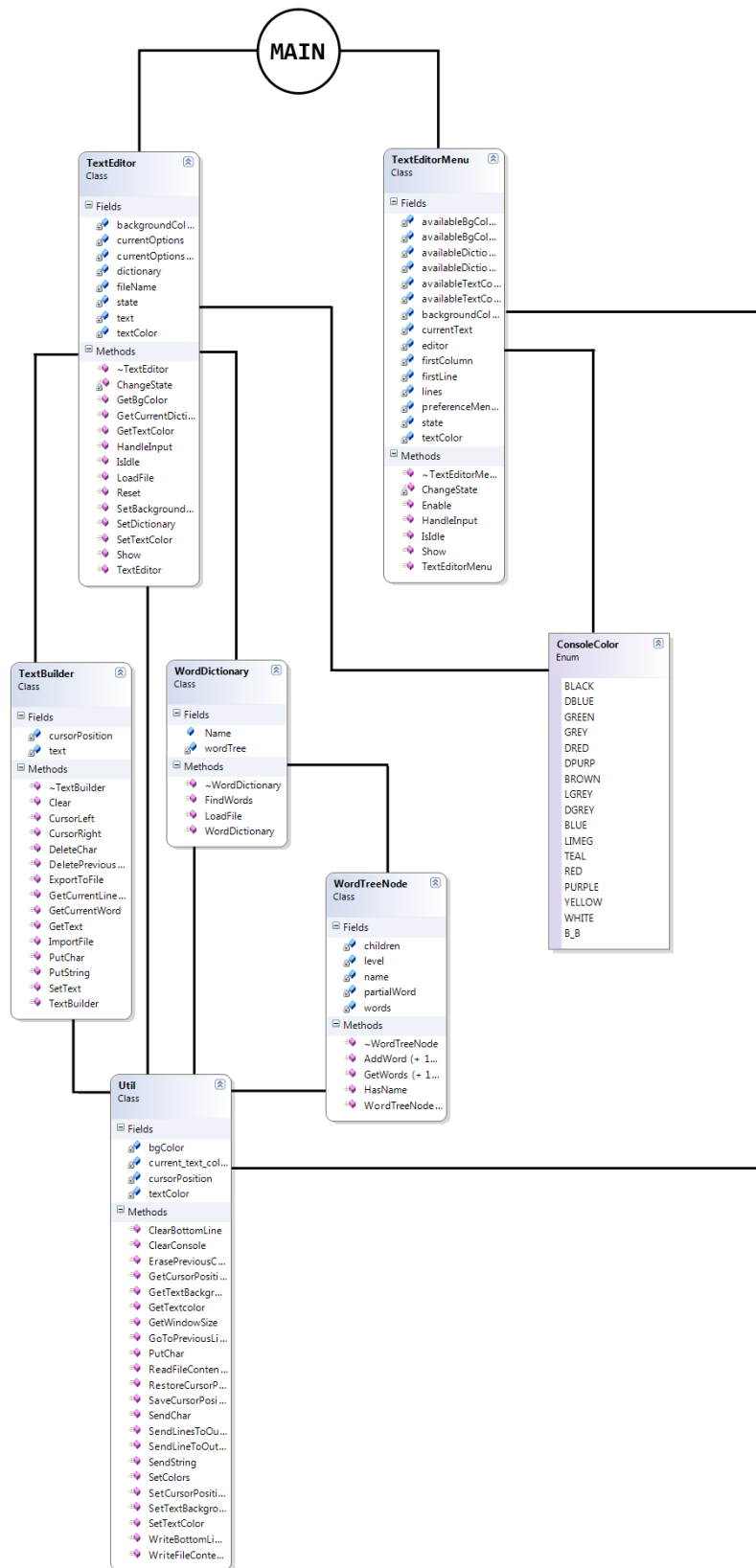


Fig. 3 - Diagrama UML

2.4 – Casos de utilização

O editor de texto poderá ser usado para leitura de ficheiros, leitura e modificação de ficheiros, e edição de ficheiros de texto.

1. Edição de novo documento: a partir do menu inicial, escolher a opção 1 e iniciar a edição do ficheiro.
2. Edição ou leitura de documento previamente guardado: a partir do menu inicial, escolher opção 2 e iniciar leitura ou edição do ficheiro.
3. Alteração do dicionário, cor de fundo e cor do texto, através do menu inicial, escolher a opção 3 e entrar no menu *Preferences*.
4. Menu *Preferences*:
 - a. Alterar Dicionário, premir tecla correspondente ao dicionário pretendido;
 - b. Alterar cor do texto, premir tecla correspondente à cor pretendida;
 - c. Alterar cor de fundo, premir tecla correspondente à cor pretendida.
5. Voltar ao menu inicial após escolha de preferências ou após edição/visualização de texto, premir tecla **ESC**.

Auto-complete, dentro do ambiente de edição de texto, ao premir a tecla **Tab** aquando da escrita de uma palavra, o editor completa a palavra se apenas existir uma palavra com esse início no dicionário, se não, apresenta um conjunto de palavras para escolha, estas palavras poderão ser seleccionadas através da tecla a elas associada que variará entre F1 e F12.

2.5 – Dificuldades encontradas

A criação do editor de texto em si e consequente edição de texto na consola foi só por si um obstáculo e ocorreu a necessidade de investigar, criar e implementar diversos protótipos até chegar a uma implementação adequada, tais como:

- a) A class Util com todas as funcionalidades de interesse para a linha de comandos como mudar cores de texto e de fundo, implementar o *backspace*, movimentação dentro do texto, entre outros;
- b) Na leitura do dicionário chegou-se à conclusão que a leitura não era feita correctamente para palavras com acento, como tal houve a necessidade de modificar o formato dos ficheiros de UTF8 para ANSI. E mudar também o tipo de estrutura em que os ficheiros eram guardados.
- c) O tempo de *load* de um dicionário é maior do que o se gostaria, não tendo sido esta dificuldade superada.

Relativamente à parte da implementação do auto-complete, foi um desafio a criação da árvore de pesquisa de modo a esta ser eficiente e com reduzida complexidade temporal.

3 – Manuais

3.1 – Manual de instalação num IDE

Abrir a pasta **CAL1011_2MIEIC3_13_TG2_4**, e fazer duplo clique no ficheiro com o nome: **Editor.sln**. Isto abrirá o Visual Studio, para poder compilar e correr deverá clicar no menu **Build**, opção **Build Solution**.

3.2 – Manual de instalação

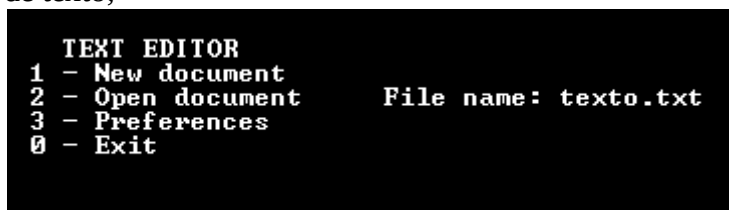
Abrir pasta **CAL1011_2MIEIC3_13_TG2_4**, abrir pasta **Debug**, duplo clique no ficheiro **Editor.exe**.

Se por algum motivo apagou a pasta ‘**CAL1011_2MIEIC3_13_TG2_4\Debug**’, necessita de compilar o programa, criando novamente esta pasta e copiar as pastas **dics** e **documents** para esta pasta.

3.3 – Manual de utilização

Imediatamente após executar a aplicação a janela da consola abrir-se-á, carregará o dicionário pré-definido (Português de Portugal), e o menu de opções do editor será visível.

1. Os ficheiros de leitura devem ser colocados na pasta ‘**CAL1011_2MIEIC3_13_TG2_4\Editor\documents**’;
2. Os ficheiros serão gravados na pasta **CAL1011_2MIEIC3_13_TG2_4\Editor\documents**’
3. Menu **Inicial**, clicando nas teclas especificadas executam-se as seguintes opções:
 - a) **New Document** – Tecla 1 – Criar novo documento de texto;
 - b) **Open Document** – Tecla 2 – Abrir o documento, previamente guardado, de texto;



```
TEXT EDITOR
1 - New document
2 - Open document      File name: texto.txt
3 - Preferences
0 - Exit
```

Fig. 4 - Abrir ficheiro

- c) **Preferences** – Tecla 3 – Abre um novo menu onde se pode escolher: dicionário, cores de fundo e do texto;
- d) **Exit** – Tecla 0 (ou Esc) – Fecha o programa.

4. Menu **Preferences**, acedido após clicar na tecla 3 no menu inicial podendo executar-se as seguintes opções:

- a) **Dictionary** – indica o dicionário actual (através do símbolo *) e permite a escolha de um dos dicionários presentes na lista, premindo a tecla correspondente ao índice do dicionário pretendido;

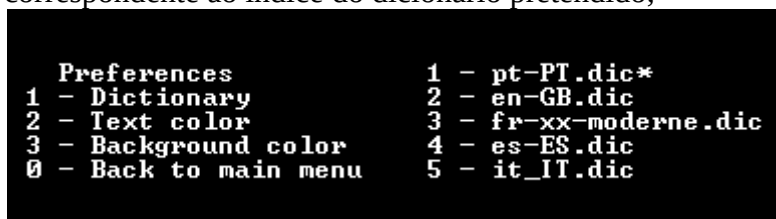


Fig. 5 - Escolha de dicionário

- b) **Text Color** – indica a cor actual do texto (através do símbolo *) e permite a escolha de uma das cores presentes na lista, premindo a tecla correspondente ao índice da cor pretendida (9 opções);
- c) **Background Color** – indica a cor actual do fundo (através do símbolo *) e permite a escolha de uma das cores presentes na lista, premindo a tecla correspondente ao índice da cor pretendida (9 opções);
- d) **Back to main menu** – Tecla 0 (ou Esc) – Volta ao menu inicial.

5. Editor

- a) Tecla ESC – Voltar ao menu inicial;
- b) Tecla CTRL+S – Gravar o documento;

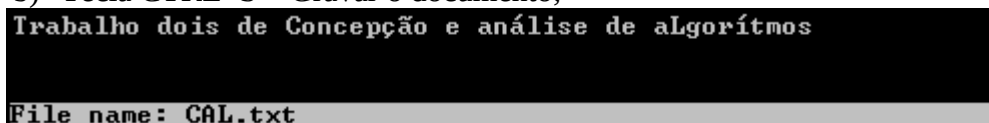


Fig. 6 - Gravar ficheiro

- c) Tecla Tab – Apresenta as palavras que completam a palavra que actual, por ordem alfabética, caso só haja uma opção preenche automaticamente essa palavra;
- d) Tecla F1-F12 – Selecção da palavra pretendida para o auto-complete de entre as possibilidades existentes.

4 – Conclusão

Foi implementado um editor de texto simples, em linha de comandos, com diversas funcionalidades de edição (gravar ficheiro, abrir ficheiro, mudar dicionário de escrita, alterar cores de fundo e de texto).

A funcionalidade mais importante, auto-complete, foi implementada com a ajuda de um algoritmo de pesquisa eficiente baseado numa árvore *N-ary tree*,.

4.1 – Desenvolvimentos futuros

Devido à possibilidade de criação de diversas funcionalidades de interface e de não existirem requisitos obrigatórios, poderão ser implementadas muito mais funcionalidades interessantes.

Seria interessante implementar a selecção de texto, tanto por rato como por SHIFT+cursor, para copiar, colar, cortar, alterar a fonte, entre outros.

Permitir o uso dos cursores para movimentação dentro do texto.

Permitir a detecção automática de dicionários adicionados à pasta dicionários de forma a inclui-los no menu de opções de dicionários do editor.

Reduzir o tempo de *load* de um dicionário.

Ou mesmo usar o MFC para a criação de uma janela (tipo *windows*) em vez do uso da linha de comandos.

Bibliografia

C++ Language Reference. (s.d.). Obtido em Maio de 2011, de MSDN:
<http://msdn.microsoft.com/en-us/library/3bstk3k5.aspx>

CPlusPlus. (s.d.). Obtido em Maio de 2011, de <http://www.cplusplus.com/>

Rossetti, R. (s.d.). *Wiki da disciplina*. Obtido em Maio de 2011, de
<http://paginas.fe.up.pt/~rossetti/rrwiki/doku.php?id=teaching:1011:cal:start>

UTF-8. (s.d.). Obtido em Maio de 2011, de <http://pt.wikipedia.org/wiki/UTF-8>